Introduction to Swirles

and Intel GPUs in general

Juliana Kwan, 6 Mar 2025

What is Swirles?

- Swirles is our new HPC cluster and the successor to Fawcett.
- There are several partitions (use `sinfo` to check):
  - The Intel components are named Cosmos XI
    - pvc: 3 nodes of Intel Data Center GPU Max 1550 (PVCs), Sapphire Rapids host
    - ~~spr: 4 nodes of Sapphire Rapids HBMs~~
  - ampere: 3 nodes of Nvidia A100
  - lovelace: 1 node of Nvidia L40
  - ~~genoa: 6 nodes of AMD EPYC 9654~~
- Intel parts funded by ExCALIBUR as a GPU testbed (open to all UK researchers)
- For now, we haven’t worked out a fair usage policy, expect this to change in the future

Access to Swirles

- Two ways to log in:
  1. ssh username@ssh.maths.cam.ac.uk and then use Slurm to queue a job or ssh into a compute node
  2. In your .ssh/config file, define a new entry:

```
Host swirles.maths.cam.ac.uk
User <username>
ProxyJump
username@ssh.maths.cam.ac.uk
IdentityFile ~/.ssh/privatekey
```

- Your username is your CRSid, the password is your usual Maths password for logging into your desktop or Fawcett
- How to check if you have a login (and how long your account is valid for):

  https://useradmin.maths.cam.ac.uk/selfservice/record

- If you don't have a login, you need to contact Maths IT with a support request: help@maths.cam.ac.uk
- We are in early access mode so it is very possible that IT support cannot fulfill your request without explicit approval from your PhD advisor/line manager.

Slurm on Swirles

How to submit jobs

- You can use mpirun directly from the node (for now…) or you can use Slurm to queue a job (recommended)
- Example Slurm script (also on the GitHub repo):

```
#!/bin/bash

#SBATCH --partition <name>        #<name> can be either pvc, ampere, genoa, spr
#SBATCH --nodes=<#nodes>          #how many nodes
#SBATCH --ntasks=<#tasks>         #how many tasks (MPI ranks) in total
#SBATCH --cpus-per-task=<#cpus>   #set to > 1 to use OpenMP threads
#SBATCH --time=<hh:mm:ss>
#SBATCH --gres=gpu:<#gpus>        #necessary for GPU resources, choose from 1 - 4

export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK

cd ${MY_WORK_DIR}
source my_modules.sh

mpirun -np <#total tasks> ./my_exec
```

Slurm on Swirles

How to submit jobs

- Can also use ‘srun’
- For example:
  - `srun -p pvc  --nodes=1 --mpi=pmi2 --cpus-per-task=28 --gres=gpu:1 <name of program>`
- If using mpirun directly on PVCs, use the option --bootstrap=fork (this will expose the tiles as separate devices).

The Swirles environment

- Swirles uses a module system, like on Fawcett.
- No modules are loaded by default (try doing a module list)
  - Some essential modules:
    - module load cmake
    - module load miniforge3/4.8.3-4-Linux-x86_64/gcc/s2ikv54u
    - module load python/3.11.7/gcc/swa4n3ak
      - Then use a virtual env to install the python packages that you need.
- The OS is Ubuntu 22.04
- Editors are vi or nano - But I have an emacs installation which you are welcome to use:

  `/alt/applic/user-maint/jk945/emacs`

  Just make an `alias` or add it to your `PATH` in `.bashrc`

Compilers on Swirles

- For an Intel oneAPI environment:
  - module load intel/compiler-intel-llvm/2024.2.0
  - module load intel/mpi/2021.13
  - module load intel/mkl/2024.2
  - The MPI compiler is then mpiicc/mpiicpx/mpiifx
- For a GNU environment:
  - module load gcc
  - module load openmpi
  - The MPI compiler is then mpicc/mpicxx/mpif90
- For CUDA:
  - module load cuda/12.6.3/gcc/bxp4amlg
  - The compiler is nvcc
  - There is no CUDA enabled MPI library yet
- NB: You can type ‘which mpirun’ to check your version of MPI

Example environment setup

On the sw-pvc0X nodes, type: `source /local/scratch/public/swirles-env.sh` to set up the following environment

```
#!/bin/bash

module load intel/compiler-intel-llvm/2024.2.0
module load intel/mpi/2021.13
module load intel/mkl/2024.2

export I_MPI_PMI_LIBRARY=/usr/lib/x86_64-linux-gnu/libpmi2.so  #this points Slurm to the API (PMI-2) for launching MPI jobs
export I_MPI_OFFLOAD=1                                         #this allows GPU to GPU direct communication via MPI (GPU aware MPI)
unset ZE_AFFINITY_MASK                                         #if this is set, then GPU pinning is automatically disabled
export ZE_FLAT_DEVICE_HIERARCHY=FLAT                           #exposes tiles as separate devices
export I_MPI_OFFLOAD_TOPOLIB=level_zero                        #set interface for GPU topology recognition to level zero
export I_MPI_OFFLOAD_PIN=1                                     #enable GPU pinning (somewhat degenerate with I_MPI_OFFLOAD)
export I_MPI_DEBUG=3                                           #this prints pinning information

alias my_sysmon='watch -n 1 /alt/applic/user-maint/jk945/pti-gpu/tools/sysmon/install/bin/sysmon'
```

This contains all the environment variables you need to run a SYCL program or use MPI enabled GPU communications or use Slurm with multiple GPUs per job

More info at Miren's guide to running GRTeclyn on Dawn: https://github.com/GRTLCollaboration/GRTeclyn/issues/67

Storage on Swirles

Where do I put my stuff?

- /home/<username> should be the same as on Fawcett/local desktop
- You should have space for outputs on /cephfs/store/<group-name> - the amount of space each group has depends on their level of contribution towards the purchase of the system.
- There is an additional 5 GB for user wide installations of software at /alt/applic/user-maint/<username>